



AFRL-HE-WP-TP-2006-0074

**Improving Human Interfaces in Military
Simulation Applications**

Steve Rowe
Joshua Band
Charles Cohen
Cybernet Systems Corporation

Michael Haas
Human Effectiveness Directorate

September 2006

Approved for public
release; distribution is
unlimited.

**Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Interface Division
Collaborative Interfaces Branch
Wright-Patterson AFB, OH 45433-7022**

20070103051

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) September 2006		2. REPORT TYPE Conference Proceedings		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE Improving Human Interfaces in Military Simulation Applications				5a. CONTRACT NUMBER FA8650-05-M-6567	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Steve Rowe * Joshua Band * Charles Cohen * Michael Haas **				5d. PROJECT NUMBER 3005	
				5e. TASK NUMBER HC	
				5f. WORK UNIT NUMBER 5R	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) * Cybernet Systems Corporation 727 Airport Blvd Ann Arbor MI 48108 ** Human Effectiveness Directorate, WPAFB, OH				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Materiel Command ** Air Force Research Laboratory Human Effectiveness Directorate Warfighter Interface Division Collaborative Interfaces Branch Wright Patterson AFB OH 45433-7022				10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/HECP	
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER AFRL-HE-WP-TP-2006-0074	
12. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited. Cleared by AFRL/PA as AFRL/WS-06-0958 on 10 April 2006.					
13. SUPPLEMENTARY NOTES This will be published in the Proceedings of Spring 2006; Simulation Interoperability Workshop.					
14. ABSTRACT This paper highlights some of the key human factors issues that we found in OneSAF, and how we chose to remedy them in our own system design. The goal is to provide guidelines for application human interface design standards so that future tools developed for the Department of Defense allow the operator to perform more effectively.					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF: Unclassified			17. LIMITATION OF ABSTRACT SAR	18. NUMBER OF PAGES 8	19a. NAME OF RESPONSIBLE PERSON Michael Haas
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (Include area code)

Improving Human Interfaces in Military Simulation Applications

Steve Rowe

Joshua Band

Charles J. Cohen, Ph.D.

Cybernet Systems Corporation

727 Airport Blvd.

Ann Arbor, MI 48108

734-668-2567

srowe@cybernet.com, jband@cybernet.com

Michael W. Haas, Ph.D.

Air Force Research Laboratory

2255 H Street

WPAFB, OH 45433

937-255-8768

Michael.haas@wpafb.af.mil

Keywords:

Human-Computer Interaction, User Interface, Simulation, OneSAF, GOMS

ABSTRACT: *Military simulation applications such as OneSAF OTB provide a flexible environment for building and simulating exercises. We recently undertook a human factors analysis of OneSAF to identify the key weaknesses in its interface, and identified several systematic problems. Among these were: excessive mouse movements, precise mouse movements, excessive visual search, finding objects in multiple data views at the same time, and repetitive, successive menu selections. Other problems exist, but these general design flaws are important in that they highlight areas of major performance loss and cognitive loading.*

These design flaws can threaten situation awareness by improperly controlling the operator's attention, making it difficult to find, interpret, and assimilate information and placing excessive cognitive demands on the operator. Careful design and analysis of information management tools using a combination of perceptual, motor and cognitive modeling techniques [1], usability engineering methods [2] and user studies can help us avoid these problems by optimizing the ability for expert users to execute tasks quickly, easily locate information, avoid confusion and recover quickly from errors. This approach to design, grounded in user and task models, leads not only to interfaces that are more aesthetically pleasing, but to tools that support better performance and are less cognitively demanding, freeing up intellectual and human resources for decision making and action.

Based on this analysis, we prototyped a number of web-based tools for improved human-system interaction that were designed to minimize these areas of performance loss. These tools were used to develop a system prototype, which we used to benchmark potential performance gains. We found these gains to be dramatic; In the case of changing a parameter of the map view, for instance, we were able to reduce the required mental operators (M in GOMS-KLM) by half and reduce mouse movement substantially as well.

This paper highlights some of the key human factors issues that we found in OneSAF, and how we chose to remedy them in our own system design. The goal is to provide guidelines for application human interface design standards so that future tools developed for the Department of Defense allow the operator to perform more effectively.

1. Introduction

The mandate from the Department of Defense is to enable joint, network-centric military operations. As we in the information technology industry develop systems to support this vision, we must mitigate the problems that arise from the over-availability of data. The authors

recently undertook the creation of an interoperability toolkit for the U.S. Air Force; we developed a tactical simulation display as a demonstration application. Since this work was being done for the Human Engineering directorate, our focus was not just on the underlying interoperability technology, but also on the design of the Human-Computer Interface (HCI). In order to

demonstrate that our system was an improvement, we needed a basis of comparison. We chose OneSAF (One Semi-Autonomous Forces), a popular constructive simulation application. We discovered many areas for potential improvement in the design of the user interface of tactical display applications. This paper discusses our methods, results, and recommendations.

2. Methods and Assumptions

The findings presented here are the result of GOMS (Goals, Operators, Methods, Selection rules) analysis [5]. This method has its roots in cognitive psychology, and has become extremely popular for characterizing human-computer interaction. A GOMS model is composed of *methods* that are used to achieve specific *goals*. The methods are then composed of *operators* at the lowest level. The operators are specific steps that a user performs and are assigned a specific execution time. If a goal can be achieved by more than one method, then *selection rules* are used to determine the proper Method.

We assume that the users of military simulation tools are experts, since GOMS analysis is only appropriate for expert-level users.

2.1 The Keystroke Level Model (KLM)

We use the KLM as the set of GOMS operators in our analysis. The KLM lends itself to the usual interaction of user with computer, as it defines the following operators: **Keystroke**, **Button press**, **Point the mouse**, **Hand** from keyboard to mouse or vice-versa, **Mental preparation** for task (including visual search), **Response time** of computer. For any method to be performed, that method is broken into a sequence of these operators. Summing the operator times yields the time needed to perform the operation. The times that we use are given in Table 1.

Table 1 - Operator Times from Kieras[1]

K	Keystroke	280 ms
P	Point mouse	1100 ms
B	Press or release mouse button	100 ms
H	Home hand to device	400 ms
M	Mental preparation	1200 ms
R	Computer Response (tool tip)	100 ms

2.2 Alternative GOMS methods

There are three other widely used variants of GOMS that we chose not to use in our work. NGOMSL (Natural GOMS Language) is a specification of the goals into a regular language. This level of formalism was not appropriate for the non-natural language OneSAF

interface. CMN-GOMS (Card, Moran, Newell) is the original version of GOMS, and does not define appropriate operators useful for our study. Finally, CPM-GOMS emphasizes user learning and cognitive load. Since we assume that all military users of OneSAF are, by definition, expert users, learning rate is beyond the scope of this particular project. Cognitive load is certainly an important factor, but such as study was beyond the scope of this initial research effort.

3. OneSAF OTB GUI

The OneSAF Objective Test Bed is a powerful application with many features. One of the drawbacks of having so many features is that it is very difficult to create an efficient, uncluttered user interface.

The graphical user interface for OneSAF is shown in Figure 1. The application was developed using the X Window system with the Motif widget set. The work area is divided into 4 sections: the top "menu and button bar" area, the left panel "high level selection" area, the bottom panel "detailed information" area, and the central "tactical map" area.

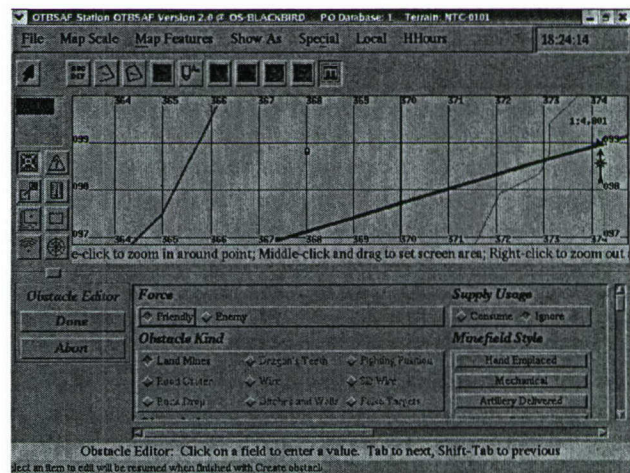


Figure 1: The OneSAF Graphical User Interface has 4 active areas on screen at all times.

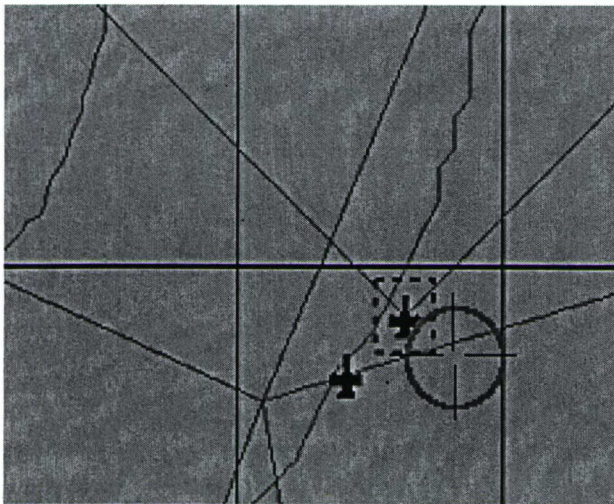


Figure 2: The hot spot of the OneSAF cursor is not the center of the crosshairs.

The overall workflow of the application is modal; that is, each of the buttons on the button bar selects a mode (such as creating new units, selecting units, modifying the map, etc.). Modal interfaces tend to decrease the operational complexity of tasks specifically in that mode. This benefit is offset by an increase in the mental operation duration because one must remember what mode the application is in, and the need to change the mode of the application in order to perform a goal from a different mode.

Another quirk of the OneSAF GUI design is that the selection cursor is displayed as a crosshair reticle, but the “hot spot” (selection point) of the cursor is actually the upper left corner of the (invisible!) square that circumscribes the crosshair (see Figure 2). This has the effect of increasing the time for all mouse-pointing operations performed by novices due to misplaced button presses. However, we find that expert users have learned to largely disregard the shape of the cursor when making selections.

4 Case Studies

This section describes our GOMS-KLM analysis of several tasks that can be performed in OneSAF and also in our prototype application.

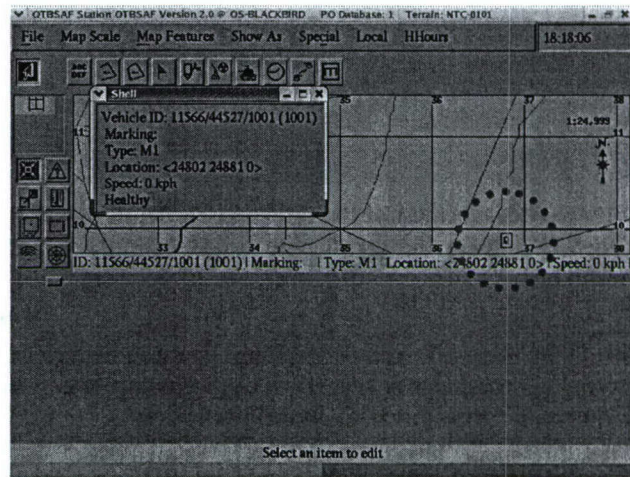


Figure 3: Middle-clicking on the unit icon (the very tiny block circled on the map) in OneSAF displays a pop-up dialog of information about the unit. Notice the large visual distance between the icon and the information box.

4.1 Determine Unit Position Information

In this case study, we examine the goal of determining the position (latitude and longitude) of a unit on the tactical display. There are two methods to do this in OneSAF:

- 1) Select the “selection” mode with the mouse (MHPBB). Middle-click on the unit (PB) to display the pop-up information window (M) as seen in Figure 3,
- 2) Select the unit in the high level selection area (MHPBB), shift focus to the information area (M), and scroll the information pane so that the location is visible (MPBPBMPBPBM).

Substituting times from Table 1 for each operation yields that method 1 predicts a time of 5.3 seconds. Method 2 predicts 12.4 seconds. The majority of the additional time in method 2 is taken by adjusting scroll bars in a sub-window (see Figure 4). Although this display allows for a substantial amount of data to be displayed, that advantage comes at the cost of a great deal of required visual search coupled with mechanical scrollbar adjustment.

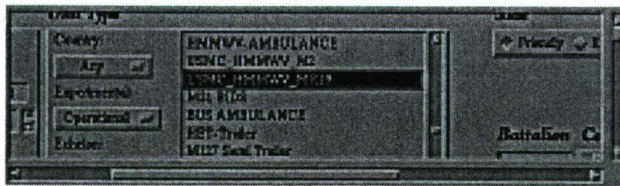


Figure 4: A close-up of the information detail panel of OneSAF. Note the scroll box nested inside another scroll box.

In future network centric warfare applications, rapid access to knowledge is key. How can we reduce the time required to access this fairly basic information?

First, often-used information should be obtainable using the “quick” method (method 1 in this example), while less frequently needed information can be relegated to a more obscure location. Focusing then, on method 1, we seek to improve performance. Consider changing the application from modal to non-modal. This alone would remove the leading MHPBB (2.9 seconds) from the operation.

Second, time can be reduced by changing the pop-up information box from one that requires a button press (0.1 seconds) to a tool-tip (configurable response time). This also has the added benefit of reducing cognitive workload.

Finally, the pop-up box should appear near the cursor to minimize visual search. This would have the effect of reducing the actual time for the mental refocus associated with the new pop-up.

In our prototype application (see Figure 5), position information is displayed by pointing the mouse cursor at the unit on the tactical display (HP) resulting in a tool tip (RM). Thus our system has a predicted time of 2.8 seconds.

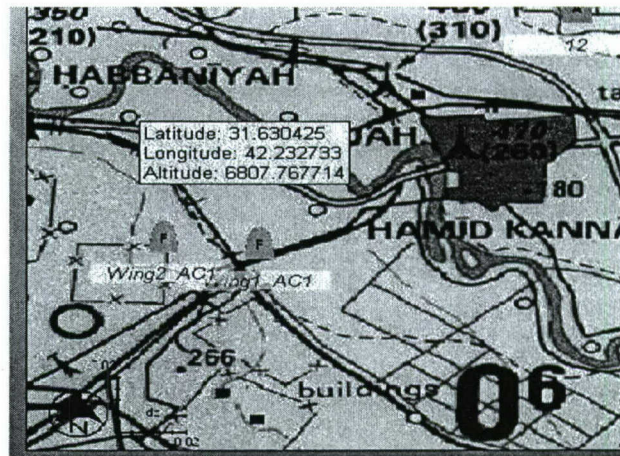


Figure 5: Our prototype uses tool-tips to display frequently-used data, thereby avoiding the need for mouse clicks or a shift in user focus.

4.2 Unit Selection

For the purposes of selecting a unit, both OneSAF and our prototype application are very similar. Both support the direct selection of the unit’s icon in the tactical display with point-and-click (MHPBB), as well as selection within an auxiliary window with point-and-click (also MHPBB). The differences in performance arise because of the required precision of the mouse pointing. As noted previously, the selection cursor in OneSAF is a cross hair with a misplaced hot spot. This increases the time it takes the user to position the mouse over the icon for selection. Moreover, if the mouse is pressed when not precisely over the icon, the effect is that the map’s zoom level is changed. Changing the zoom level can move the icon of interest completely out of the user’s field of view, requiring remedial action to restore the workflow.

Using the static unit selection window is easier, but it requires the user to know which unit in the window corresponds with the unit of interest in the tactical display. In OneSAF, this is accomplished solely through unit designation. In order to find the unit in the unit selection window that corresponds to a given icon on the tactical map, the user identifies the unit name (M), and then performs a visual search through the unit selection window (M, and if scrolling is required, HPBPMB).

Although our results were based on a simulation application interface, it should be clear that future joint forces command and control applications will have a similar set of functionality, and therefore a similar user interface. In order to enhance the effectiveness of our commanders using such software, we must create standards with the provision for *engineering* user interfaces to minimize user cognitive load and eliminate wasted physical motion.

7. References

- [1] John, B. E., & Kieras, D. E. (1996). "Using GOMS for user interface design and evaluation: Which technique?," *ACM Transactions on Computer-Human Interaction*, 3, 287-319.
- [2] Nielsen, J. & Molich, R. (1990). Heuristic Evaluation of User Interfaces. *Computer Human Interface (CHI) 90 Proceedings*, 249-255.
- [3] Wickens, C., & Hollands, J.: *Engineering Psychology and Human Performance*, Prentice Hall, New Jersey 2000.
- [4] Shneiderman, Ben: *Designing the User Interface*, Addison Wesley, Reading, Massachusetts 1998.
- [5] Card, S., & Moran, T, & Newell, A.: *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Associates, Inc., New Jersey 1983.
- [6] Helander, M., & Landauer, T., & Prabhu, P.: *Handbook of Human-Computer Interaction*, North Holland, Amsterdam 1997.
- [7] Brinck, T., & Gergle, D., & Wood, S.: *Designing Web Sites that Work – Usability for the Web*, Morgan Kaufmann Publishers, San Francisco 2002.
- [8] Ware, Colin: *Information Visualization*, Morgan Kaufmann Publishers, San Francisco 2004.
- [9] Beyer, H., & Holtzblatt, K.: *Contextual Design*, Morgan Kaufmann Publishers, San Francisco 1998.
- [10] John, B., & Prevas, K., & Salvucci, D., & Koedinger, K.: "Predictive Human Performance Modeling Made Easy," *CHI 2004*, April 24–29, 2004, Vienna, Austria.

Author Biographies

STEVE ROWE is a senior software engineer at Cybernet Systems Corporation. Besides being well versed in simulation environment, intelligent agents, and scenario development work, he has taught in the classroom and also worked in the development of automated teaching materials at KnowledgeNet Corporation.

JOSHUA BAND is a research engineer at Cybernet Systems Corporation. He has significant experience

developing massive multi-user networking and simulation environment technologies. He acted as lead interface engineer on the prototype testing application mentioned in this paper.

DR. CHARLES COHEN has been working in the fields of image processing, robotics, human-computer interaction, and artificial intelligence for over a decade. At Cybernet, he has been the project manager for many projects for the United States Armed Forces (Air Force, Navy, and Army), National Aeronautics and Space Administration, and other government agencies. His projects include work on simulation, training, real-time optical pose determination, robotics, virtual reality, object identification, feature and body tracking, and human performance evaluation. Dr. Cohen's main areas of interest are gesture recognition and massive number of agent network architectures and systems, which he is developing and integrating for the Army and Air Force.

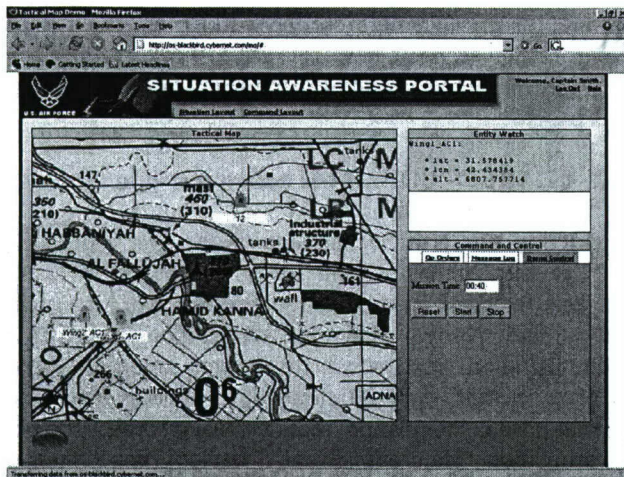


Figure 6: Highlighting a unit's icon in our prototype application highlights the corresponding entry on the right in bright yellow.

In our system, a similar visual search is required. However, we reduce the time required for the visual search by reducing the level of cognition required. We do this by simultaneously highlighting a selected icon with the corresponding display in the unit selection window (see Figure 6). This reduces the time it takes the user to recognize the unit of interest, because it is visually set apart from the other units in a very dramatic fashion.

Another way to augment this operation would be to automatically scroll the unit selection window such that the selected unit is visible.

5. Decreasing the Mental Operation Times

In this section, we depart from formal GOMS analysis and discuss some general improvements that can be made to the OneSAF user interface based on qualitative observation.

5.1 Modality

As noted in section 3, OneSAF uses a modal operation paradigm. Without debating the wisdom of the choice of a modal interface, we believe that the efficiency can be improved by using the tabbed pane user interface widget. This has the effect of changing the current look of the interface sufficiently to constantly remind the user which mode they are in, thus reducing cognitive workload.

We demonstrate a tabbed interface (Figure 7) for different modes of operation in our prototype application. The tabs provide access to controls for reviewing operational orders, playing back simulation data, and scanning messages.

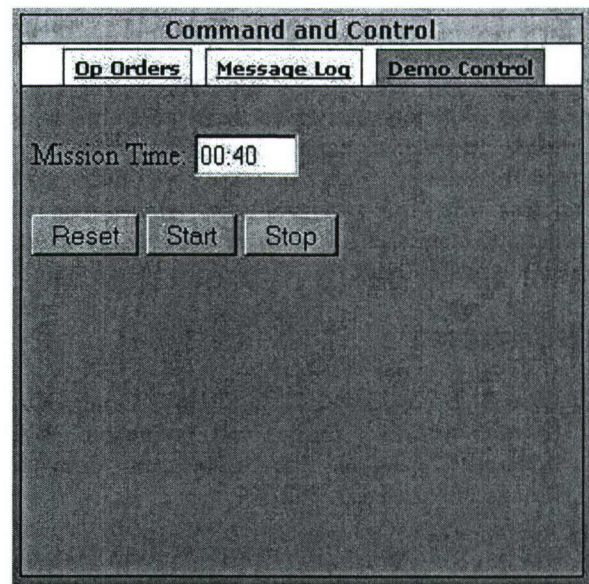


Figure 7: Use of tabbed pane to localize the changes of a modal interface.

5.2 Visual Search

The time it takes for a user to find the next interface element with which to interact is also part of the mental operation equation. If an element is always displayed in the same place, an experienced user will very quickly move their eyes to that place; in this case, there really is no "search". Similarly, if an element appears in a place that is very near the users current focus, then the eyes do not need to move, and again no search needs to take place.

OneSAF's interface requires the user to frequently move between the four control areas of the interface. Each such shift requires a visual search (sometimes accompanied by the use of scroll bars) in order to continue the operation.

In our application, we chose to use context-sensitive pop-up menus (that appear next to the mouse icon) for most of our control interfaces. This has the effect of keeping the controls where the user's attention is, rather than forcing them to go find the controls.

6. Conclusion

We have provided some case studies of the performance improvements possible with OneSAF. We have introduced the GOMS-KLM method for analyzing user interface performance trade-offs. We have shown that improvements can be made by both reducing the number of operations required for a method, and by reducing the factor that an operation (e.g. Mental) actually costs.